

# Discrete event simulation for supporting production planning and scheduling decisions in digital factories

B. Kádár<sup>1</sup>, A. Pfeiffer<sup>1</sup>, L. Monostori<sup>1,2</sup>

<sup>1</sup>Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest, Hungary

<sup>2</sup>Department of Production Informatics, Management and Control, Faculty of Mechanical Engineering, Budapest University of Technology and Economics, Budapest, Hungary

## Abstract

The paper tackles the problem of managing uncertainties during the execution of predictive schedules in a dynamic environment. The dynamic environment in question is represented by a simulation model which constitutes a coherent part of a Digital Factory solution. The model is connected to an integrated production planner and job-shop scheduler system with flexible modelling capabilities and powerful, scalable solution methods. The paper addresses the simulation module of the architecture highlighting its main functionalities. The paper also shows the potential of using the simulation model in two different ways. Both applications support the production planning and scheduling decision making process, but from two different viewpoints. On the one hand, the model presented in the paper can be applied as a schedule evaluator, on the other, it can serve as a simulation-based scheduler as well. A brief description about possible schedule evaluation criteria is also provided. Within the integrated hierarchical architecture the schedules are calculated by a constraint-based deterministic scheduling algorithm. Results of experiments which were achieved by using the model of a real, large job-shop environment are also provided.

## Keywords:

Digital Factory, Simulation, Constraint-based scheduling, Schedule evaluation, Emulation

## 1 INTRODUCTION

The concept of the Digital Factory, i.e., the mapping of all the important elements of the enterprise processes by means of IT provides a unique way for managing the problems, which enterprises face in today's changing environment. According to [1], the Digital Factory concept can be understood as an approach for an improvement in handling, managing and control of changes in a production system. With the power provided by the IT components of the Digital Factory plans of higher quality can be generated. Additionally, the concept provides better support in the handling and execution of planning processes

Among other things, the Digital Factory concept:

- enables the integrated handling of the data on the products, processes and resources, furthermore, the systematic organisation of the manufacturing knowledge,
- enables planning by models and provides a harmonised combination of sub-models built from different planning aspects,
- allows the evaluation of designing and manufacturing activities based on precise computer simulation before starting manufacturing,

In a broader sense the Digital Factory concept can be regarded as an integrated, synthetic manufacturing environment to enhance all the levels of decision and control.

### 1.1 Simulation

Simulation can be considered as one of the technologies used in the Digital Factory concept. This is a powerful tool

often applied to the design and analysis of complex systems. Decisions are made about the system by constructing its computer models and experimenting on the models. In order to construct valid models of complex systems (e.g. manufacturing, transport, service systems etc.) and their processes, the models should represent the discrete event evolution of the system, as well as the features of the underlying continuous processes.

The realisation of a simulation is a cyclical and evolutionary process. The first draft of the model will frequently be altered to make use of in-between results and, in general, the final model can only be elaborated after several cycles. Building a model is rarely an end in itself. The goal of most analyses is to be able to make a 'good' decision. Whether the system is a production line, a distribution network or a communication system, we can use modelling for gaining knowledge of the system at different life-cycle phases, evaluating a certain feature in the system, making prediction on system performance, comparing several alternatives, detecting problems and for evaluating and improving system performance. Simulation results help to define the physical layout of a system, its operating limits and control system. Models are applied as a basis for extensive experimentation, often using automatic procedures to determine optimal or robust solutions.

The features provided by the new generation of simulation software facilitate the integration of the simulation models with the production planning and scheduling systems. Additionally, if the simulation system is combined with the production database of the enterprise, it is possible to instantly update the parameters in the model and use the simulation parallel to the real manufacturing system

supporting and/or reinforcing the decisions on the shop-floor.

The paper illustrates a simulation framework that supports decision making process in production planning and scheduling (PPS). It is coupled in a hierarchical planning and scheduling system and enables the testing and evaluation of deterministically calculated advance schedules. The overall architecture provides the base of a dynamic scheduling system which assists both reactive and proactive scheduling decisions.

## 1.2 New approaches to apply simulation in production systems modeling

This section describes the possible applications of simulation on the different levels of a production system. The different roles of simulation in production planning and scheduling as well as in production control systems is shown in Figure 1.

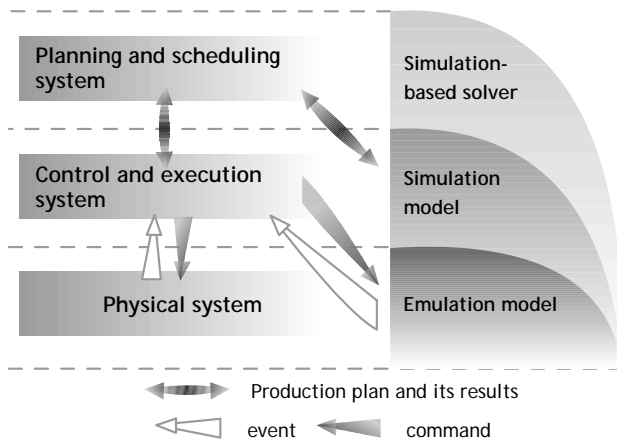


Figure 1: The possible roles of simulation in PPS systems.

To make the categorization easier three main levels are defined. A real production environment is presented on the left side of the figure. The physical system constitutes the lowest level that includes the real manufacturing facilities of the factory.

The middle level corresponds to the control and schedule execution system. Generally, this is the Manufacturing Execution System (MES) of the production system. It controls the physical system, i.e., propagates the scheduled tasks as commands to the physical system and receives reports about the execution state of the plan. This level, generally, does not have any complex planning or decisions-making function but a close connection to the resources at a lower level. Any change in the state of the lowest level is described by events, and these events will cause reactions in the control system.

The highest level represents the integrated planning and scheduling system where complex decision-making and scheduling processes are carried out. The plan is executed by the physical system under the control of the second level. The planning and scheduling system gets feedback information about the plan from the second level. Both, the new planning and scheduling tasks and feedback information are received from the production database. With regard to production systems, the third level is usually very complex. As described in [2], these systems are tested on the shop-floor after the installation only, which results in costly failures at the start-up stage. In order to eliminate the technical problems in the design phase, the modelling and simulation of the whole system is needed. However, in order to model the three levels in one framework, substantial compromise is needed. A good

solution is to separate the model of the systems, in the same way as in reality, as represented on the right side of Figure 1.

Generally, a simulation model is developed, for modelling the overall behaviour of the system, including control methods and reflecting the physical system by modelling the resources. Mainly this kind of simulation model (**simulation model** in Figure 1) is applied for testing and validating production plans and collecting statistical data. The details, the granularity and the time-horizon of the simulation model depend on the system to be modelled. These features should be chosen in a way that they should enable fast simulation runs, ensuring a great number of model runs, which gives statistical confidence.

Expanding the simulation with additional components (e.g. optimization algorithms) powerful simulation-based solvers can be created that may be applied in the solution of planning and scheduling problems (**simulation-based solver** in Figure 1). Generally, in a system like this, the simulation module is applied as an evaluation (fitness) function of an optimization algorithm. These algorithms may reside outside the simulation software in a separate solver system or in the simulation system as an integrated sub-module.

In contrast to simulation, emulation reflects only the state of the underlying production system. Emulation (**emulation model** in Figure 1) is actually a simulation model without control inside. This differs from the typical discrete event simulation models, but the applied modelling techniques are the same. Instead of validating production plans, emulation is applied for testing and evaluating control systems. Emulation models are used in a much more precisely defined way; in order to test the operation of the control system under different system loading conditions, and as a risk-free means of training system operators and maintenance staff. Emulation and simulation models are used for experimentation in a different way. Emulation reflects more precisely the system that will be implemented, and as such, can be used to carry out a constrained series of verification procedures to ensure the performance or reaction of the control system [3]. Emulation may reduce the developing time of control systems and shortening this way the time-to-market, furthermore, allows testing of control systems faster than it is done in real-time and under safe conditions. The conditions under which the tests are carried out can be better controlled, allowing the study of different scenarios the control system has to deal with. The effects of worst-case scenarios and machine break-downs can easily be studied.

The simulation system, which will be detailed in the remaining part of the paper, is classified into the middle layer in Figure 1 and will be used as a schedule evaluator for the highest level.

## 2 SCHEDULING AND SCHEDULE EVALUATION

The broad goal of manufacturing operation management, such as a resource constrained scheduling problem, is to achieve a co-ordinated efficient behaviour of manufacturing in servicing production demands, while responding to changes on shop-floors rapidly and in a cost effective manner. Operation scheduling is viewed as a major issue which is a complex task requiring co-ordination. Shop-floor scheduling, such as resource constrained scheduling problems in general, is a combinatorially complex, NP-hard problem, thus is unfeasible to be solved computationally by the sole use of conventional Operations Research (OR) approaches. Artificial Intelligence (AI) based or hybrid techniques using

domain specific heuristics are necessary to guide the search and to provide satisfactory solutions in due time. This demand put the constraint satisfaction techniques in front [4].

The quality of factory scheduling, generally, has a profound effect on the overall factory performance. The advanced generation of factory schedules is necessary to co-ordinate the manufacturing activities in order to meet organizational objectives, and to anticipate potential performance obstacles (e.g., resource contention), thus to minimize the disturbing effects on the overall manufacturing system operation. In industrial practice, however, at least two factors confound the use of predictive (advance) schedules as operational guidance [5], [6]:

- Advance or predictive schedules result from scheduling systems running with static models that ignore important new operating constraints/ objectives of live shop operation which correspond to the live status of executed processes and the data resulting from their real-time monitoring.
- They cannot cope with the environmental and internal uncertainties such as unexpected production demands raised by changing market conditions, late deliveries, failed operations/break-downs of machines/equipment, unavailability of operators all of which work against efforts to follow predictive schedules.

Traditionally, research on scheduling concentrates on off-line scheduling problems. However, in reality, shop-floor scheduling problems, are of dynamic nature, which necessitates more complex techniques. The closer we are to the realization of plans and schedules, the higher the chance of unexpected events is that may render plans and schedules inadequate. That is why practical scheduling is driven by uncertainty, and the methods applied in dynamic job-shops rarely utilize theoretical results [7].

The performance of a shop-floor control system depends mostly on its ability to rapidly adapt schedules to current circumstances. Scheduling techniques addressing the dynamic scheduling problem are called dynamic scheduling algorithms. These algorithms can be further classified as reactive and proactive scheduling techniques. An additional categorization of scheduling techniques relates to the stochastic or deterministic characteristics of the problem [6], [8], [9].

Reactive scheduling is, generally, conceived as a real-time revision or repair of a complete but execution-time flawed schedule to keep in line with the live status of shop-floor processes and events and to make it further executable. In addition, the importance of the stable behaviour of manufacturing system operation has been recognized. Operational solutions for reactive scheduling mean complete rescheduling, deferred commitment and tweaking [8]. Some probabilistic representations of scheduling uncertainty have been reported on in [10], [11]. Reference [12] gives a detailed survey on how other possible approaches were applied in scheduling.

## 2.1 Evaluation of schedules

The quality of factory scheduling, generally, has a profound effect on the overall factory performance. As stated in [13], an important aspect of the schedule measurement problem is whether an individual schedule or a group of schedules is evaluated. Individual schedules are evaluated to measure its individual performance. For a predictive schedule, the result may determine whether it will be implemented or not.

There might be different reasons for evaluating a group of schedules. One of them is to compare the performance of the algorithms with which the different schedules were calculated. The comparison of different schedule instances

against different performance measures is an other option in the evaluation of a set of schedules for the same problem. According to [13], relative comparison assumes that for the same initial factory state two or more schedules are available, and the task is to decide which is better. The task is to decide which one is better from two schedules, or which one is the best from a group of schedules generates additional questions. In a complex manufacturing environment it is probable that different schedules will perform better against different performance measures. Therefore, the selection of the best schedule will always depend on the selected performance measure(s) and thus, on the external constraints posed by the management of the enterprise.

An absolute measurement of schedule quality consists in taking a particular schedule on its own and deciding how „good“ it is [13]. This requires some set of criteria or benchmarks against which to measure.

Regarding the predictive schedules, a set of decisions is made on the base of estimates on future events, without knowing the actual realizations of the events in question until they actually occur. Taking this fact into consideration, Kempf et. al [13] differentiate between the static and dynamic measurements of predictive schedules. A static measurement means the evaluation of the schedule independently of the execution environment.

Contrary to static measurement, the dynamic measurement of a predictive schedule is more difficult. In this case, beyond the static quality of the schedule, the robustness of the schedule against uncertainties in the system should also be taken into consideration.

Another aspect in the evaluation of schedules is the state of the manufacturing system after the execution of the schedule. In Kempf et. al [13] these parameters are compared as state measurements, which evaluate the end effects of the schedule at the end of the schedule horizon.

Regarding the evaluation classes listed above, a dynamic measurement of individual predictive schedules will be presented in the following sections. The schedules are computed by a constraint-based scheduler and afterwards they are evaluated in a simulation environment that imitates the dynamic behaviour of the real production system. The maximum and average tardiness of all jobs in the system over the scheduled horizon (1 week) and the number of unprocessed jobs are applied in course of the evaluation.

## 2.2 Simulation model as a schedule evaluator

Simulation captures the relevant aspects of the PPS problem, which cannot be represented in a deterministic, constraint-based optimization model. The most important issues in this respect are uncertain availability of resource, uncertain processing times, uncertain quality of raw material, and insertion of conditional operations into the technological routings.

Here the developed simulator is utilized as a component of a higher level system taking the role of the real production system and acting as a "quasi emulator". In quasi-emulation the simulation takes the role of the MES and real production system. The control logic of the simulator ensures that the schedule be executed in the same way as it was passed by the short-term scheduler (Figure 2)<sup>1</sup>.

The reason of the intention to connect the scheduler to a discrete event simulator was twofold. On the one hand, it serves as a benchmarking system to evaluate the schedules on a richer model, on the other hand, it covers the non-deterministic character of the real-life production

<sup>1</sup> In a true *emulation mode* the simulation model would take the role of the real production system only.

environment. Additionally, in the planning phase it is expected that the statistical analysis of schedules should help to improve the execution and support the scheduler during the calculation of further schedules. The evaluation of schedules is measured over several runs of the discrete event simulator where the number of replications (independent simulation runs, with different random numbers) depends on the construction of confidence intervals.

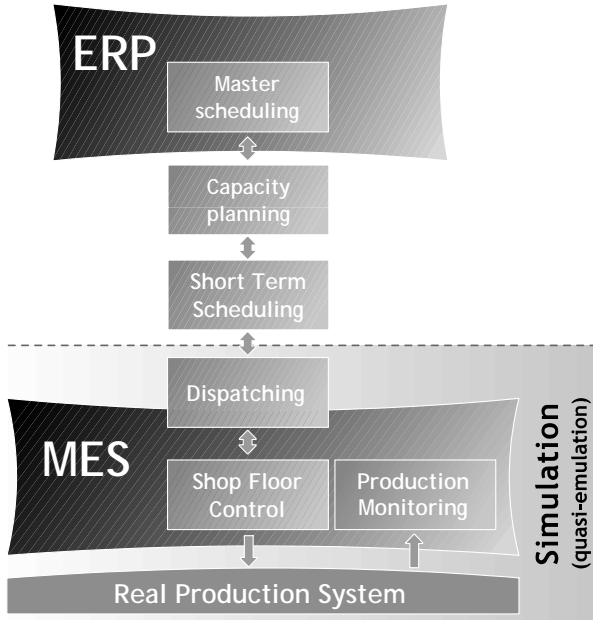


Figure 2: Applying the simulation model as a *quasi-emulation* module for schedule evaluation.

The main functions of the discrete event simulator are as follows: it

- evaluates the robustness of daily schedules against the uncertainties,
- helps in visualizing and verifying the results of the PPS system,
- supports the systematic test of the pilot PPS system,
- offers a benchmark platform for the calculated schedules,
- supports dynamic rescheduling decisions.

### 3 ARCHITECTURE OF THE DEVELOPED PRODUCTION PLANNING AND SCHEDULING SYSTEM

Based on previous explanatory experiments and basic research, a multi-tiered system structure was defined. The layers of the system are as follows:

- The solution of medium-term, integrated capacity and production planning problem is provided by an integer-linear programming approach. (**Capacity planning** in Figure 2)
- The solution of the short term, detailed finite scheduling problem is calculated by a constraint programming technique. (**Short term scheduling** in Figure 2)
- The evaluation and analysis of the predictive short term schedules is carried out by a discrete event simulation model. (**Simulation quasi-emulation** in Figure 2)

An important practical requirement is that the system components should be able to work with the data stored in existing production information systems. The details of the capacity planning module and the finite capacity scheduler are described in [14], [4], [15].

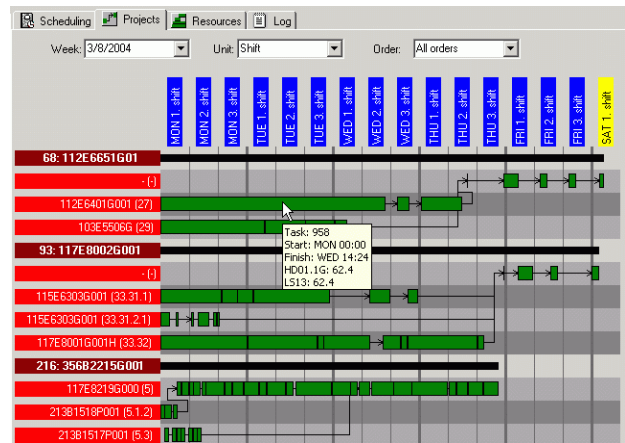


Figure 3: User Interface of the short-term schedulers.

Figure 3 shows a user interface screenshot of the short-term scheduler, with a detailed schedule plan.

In the following sections the simulation module of the above architecture and the schedule evaluation approach are described.

#### 3.1 Architecture of the simulation module

The main requirements for the simulation module are as follows:

- Common data, on-line and bi-directional connection to the scheduler,
- Support for Input/Output inspections,
- Support for different playback strategies,
- Playback time horizon: 1 week,
- Short response time, making multiple model runs possible.

In order to meet all the requirements for a flexible simulation system, the structure presented in Figure 4 has been developed. Simulation and finite capacity job-shop scheduler has the same production database. Resources, products, process plans, production information, etc. are transformed exactly to the same form for all system components. Hereby, the complexity of integrating the simulation module into the system is significantly reduced. None the less, the common data tables ensure data integrity during the creation of the simulation; moreover, the data-model serves as a basis for the more detailed shop-floor model. Running the simulation by applying the basic data tables results in a waste number of queries during the model run, reducing the simulation speed significantly. However, in order to ensure enough number of simulation replications for the evaluation of a short time production schedule, the total response time should be minimized. To resolve the above two contradictory objectives an exhaustive data pre-processing phase is included in the simulation process.

The data-processing is carried out before the overall simulation (phase **a** in Figure 4). The redundant data storage in the simulation model is compensated by the advantage of the shorter response time.

Modelling real production systems frequently brings up the problem of handling hundreds of resources in a simulation model. Having the modelling objects in hand, which were created on the base of the conceptual model, in our architecture the simulation model is created automatically based on the pre-processed data (phase **b** in Figure 4).

The automatic generation of the model is followed by the initialization phase (phase **c** in Figure 4). In this phase, besides classical parameter settings, the procedure involves the generation of input-parameter-specific model

components (entities such as products, operators). Contrary to the previous phase, this one is carried out for each replication.

The simulation runs are repeated until the required number of replications is obtained (phase **d** in Figure 4). Each replication is a terminating, non-transient simulation run, having the same initial parameters and settings, but different parameters for uncertain simulation times and events generated on the base of random numbers. In the last phase the plan is evaluated by using the evaluation criteria and the results of the evaluation process are interpreted by shop-floor managers who are predestined to take necessary actions (the shop-floor manager, phase **e** in Figure 4).

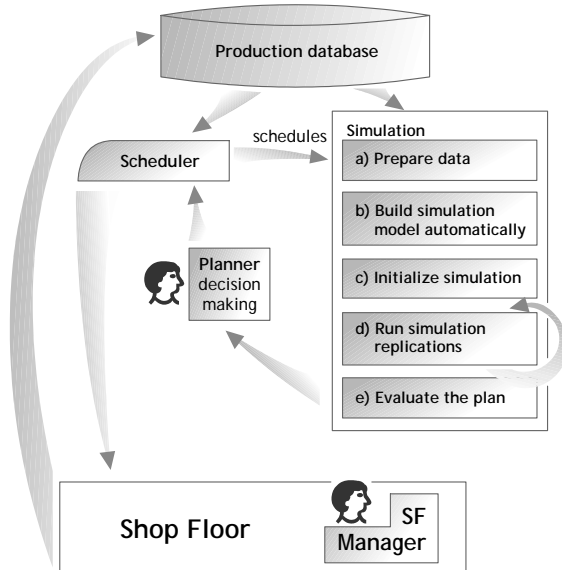


Figure 4: Architecture and the main process flow in the simulation module.

#### Uncertainties in the simulation model

The basic types of uncertainties modelled in the simulation model are as follows:

- Downtimes: due to failures and/or the unexpected absence of machines and/or workers,
- Processing time: the actual processing time of some operations may depend on the proficiency and skill of the worker. Processing times may be shorter or longer than planned,
- Rework and adjustment: the execution of specific operations depends on the result of quality check operations. Based on the result of the check, they may be repeated or some adjustment operations are performed.

#### Execution of schedules

Because the time horizon is limited and the orders under process are included in the schedule in question at the initialization, the system does not have a transient phase coming from the warm-up period. This is handled on the level of the short term scheduler that determines the optimal schedule taking the current work in-process (WIP) state of the shop-floor into account.

In order to reduce the rigidity of the schedule during execution, the fixed start times of operations are removed and only the sequence of the operations on the various resources are kept. We use this control rule in order to follow the predictive schedule as far as possible. By default, an operation can be processed if it is in the front of all of its queues. However, since there are not only single, but also alternative resources, we may apply a relatively

liberal execution policy, while keeping the consistency of the overall job-shop schedule. Accordingly, an operation may be processed any time if it does not cause lateness in the subsequent operations. As a main principle, the simulator should play back the plan only without changing the optimized sequence of the tasks.

A case-study with several experimental results is summarized in the following section. The model of the case-study represents a real, large job-shop environment.

## 4 CASE-STUDY

A case-study was elaborated at a factory that produces mechanical products by using machining and welding resources, assembly and inspection stations and some highly specialized machines. Production is performed in a make-to-order manner where deadline observance is an absolute must, even regarding unpredicted orders. Since quality assurance is a key issue, tests may result in extra adjustment operations. The planning and scheduling method was validated and tested with the real-life data. First, projects were generated from existing routing tables and Bill of Materials (BOMs), then, using the resource calendars, the planning problem was solved on a 15-week horizon, with a time unit of one week. Then, the production plan was passed to the constraint-based finite job-shop scheduler that worked with a 10 min. time unit.

The object-oriented hierarchical simulation model of the plant is based on the functional decomposition approach. The simulation includes the modelled elements of the real plant and each unit of a production set is identified uniquely and traced during its lifecycle. The simulation model is created following the simulation modelling process described above. The deterministic inputs of the simulation are provided in three main data tables. These are tables of resources, process plans, and the short-time schedule, passed by the scheduler.

The resources of the plant are categorized in two main groups: machine and personnel. The stochastic inputs are represented by the uncertainty parameters mentioned above. Based on the resources table, the whole model is generated automatically during the data preparation phase. This is combined with the weekly calendar of the resources.

The simulation model of the case-study implements a dual-frame architecture (Model and SimManager frames). SimManager is responsible for the data preparation, model creation, initialization and evaluation. The components of the model are created into the Model frame. One of the main advantages of this server-client architecture is that it makes the transfer of the application to a distributed simulation environment possible.

Input	Size
Number of tasks in a job	20-500
Working resources/week	80-120
Average number of jobs/week	15-20
Average number of tasks/week	1500-2000

Table 1: The parameters of the scheduling problem in the case-study.

The shop-floor of the case-study includes more than 100 resources, all of which are modelled in the simulation module. The short-term schedule table contains approx. 2000 tasks to be executed in one replication. The time frame of one simulation replication is one week. The statistical data are collected both on the resource and product sides. Figure 5 shows the developed simulation model. Table 1 summarizes the size of the case-study scheduling problem.



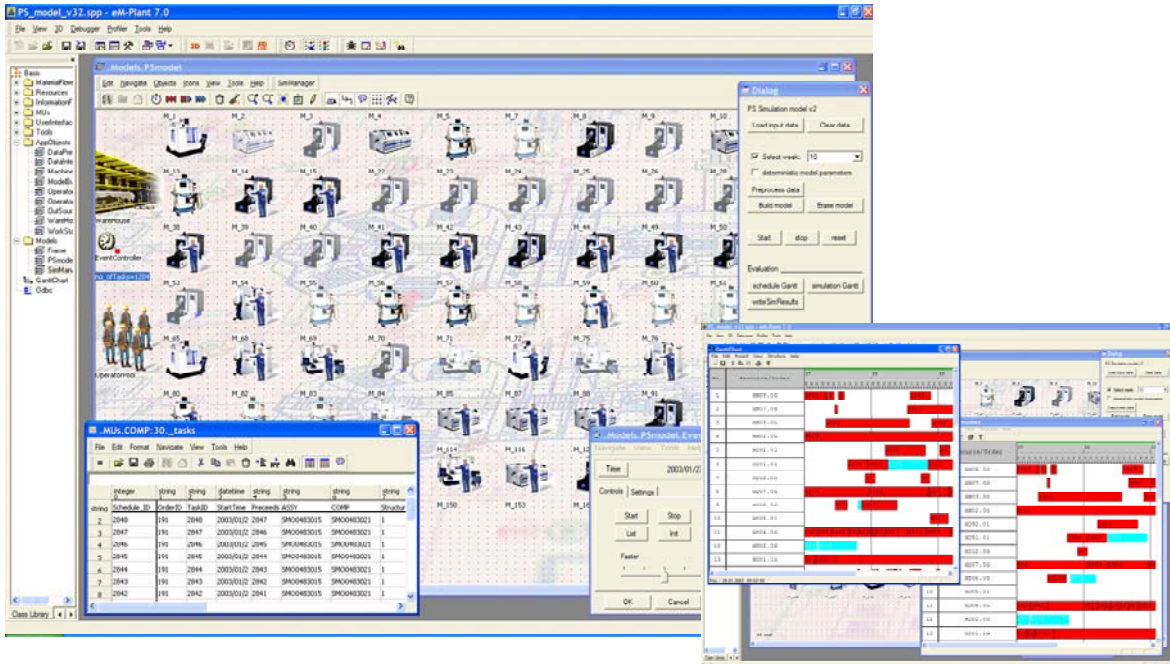


Figure 5: The visualized simulation model of the case-study and the Gantt charts of the initial and the executed schedule, respectively.

The initial schedule and the schedules after the simulation runs can be visualized in the simulation module providing a comfortable user interface for necessary interventions (Figure 5).

#### 4.1 Results of the evaluation

In the case-study the experimental simulation runs investigated the effect of the following three uncertainty factors:

- Availability of the machines, which may range from 90% to 99%.
- Variable processing times, depending on the state of the production facilities of the shop floor and the skills of the workers.
- Variable number of the workers with the same skills in different groups.

The most important objective in the factory of the case-study is the minimization of tardy jobs and WIP level. Additionally, the simulation studies always have a one-week time horizon. Taking these facts into consideration, mean tardiness, maximum tardiness and the remaining WIP level after the schedule execution (after the executed week) are considered as responses and performance measures in the evaluation of the schedules.

Table 2 shows how the value of average tardiness changes in deterministic and stochastic cases.

Applied play-back strategy	Average tardiness (h)	Maximal tard. (h)	WIP
Deterministic process times	-	0	0
Stochastic process times	2.74	17.13	5
95% machine availability	5.25	18.65	27

Table 2: Illustrative results of deterministic and stochastic schedule execution procedures for one-week (average values in hours, calculated from 250 simulation replications).

Deterministic execution means that no uncertainty was set in the simulation. As expected, in this case the executed schedule is exactly the same as the planned one. In the stochastic processing time scenario (row 2 in Table 2) the processing times of the tasks are set randomly, supposing uniform distribution. The lower bound is 90%, while the upper one is 130% of the planned process time. This set-up includes the variation of processing times coming from differences in the skills of the operators, as well. Row 3 in Table 2 refers to the machine availability which is 95% in this experiment. To create a real dynamic scheduler, after the simulation the executed schedules are updated in the common database. The tasks which were not executed are added to the plan of the next week during the rescheduling process. The average time of one simulation run is approximately 10 seconds.

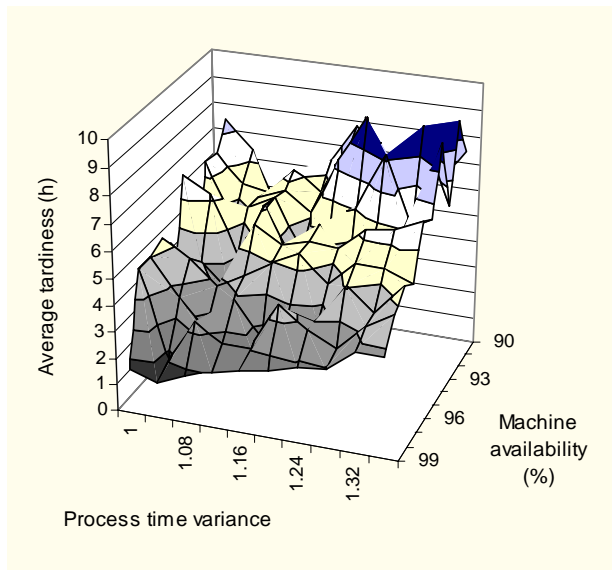


Figure 6: The dual effect of machine availability and processing time variance on average tardiness.

### Integrated effect of two factors

Figure 6 shows the effect of both machine availability and processing time variance on average tardiness. Apart from the fact that the chart reinforces the prior expectations about the average tardiness effect of input values from different interval sets can be analysed together.

### The effect of missing operators

Figure 7 represents the effect of weekly operator availability on average tardiness value for one selected schedule. The dark bars show the results where the number of operators per group was decreased by 10% while the white bars represent the results with 20% less operator per group. The replications were carried out sequentially group by group, analysing the effect of only one group at once. The results of the experiment show that groups 7 and 8 have the main effect on the average tardiness. The other operator groups have no significant influence on the same output value. Results were calculated from 20 different parameter settings, each with 10 replications.

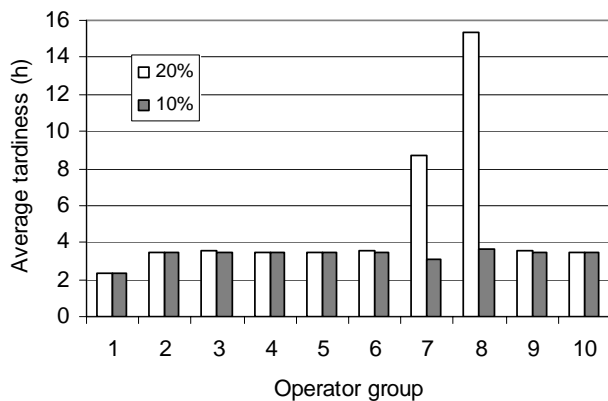


Figure 7: The effect of missing operators.

### The effect of new employees with decreased efficiency

Experiments were carried out for the evaluation of different worker groups including operators with different skills. Supposing that new operators are employed and during the "learning period" their efficiency is smaller, we investigated their overall effect on the planned schedule.

Group	SHIFT1		SHIFT2		SHIFT3	
	No of op.	No of new empl.	No of op.	No of new empl.	No of op.	No of new empl.
1	2	1	2	1	2	0
2	13	4	13	2	13	1
3	18	4	21	5	17	3
4	3	1	4	2	3	0
5	4	1	4	1	4	1
6	4	2	4	1	4	1
7	8	2	10	3	9	1
8	12	3	15	3	13	2
9	8	2	10	3	8	2
10	3	1	4	1	3	0

Table 3: Input data table for sensitivity analysis on operator efficiency level.

Table 3 shows the input settings of the experiment. There are 10 different operator groups. Column "No of op." represents the total number of the workers in a group, while column "No. of new employees" shows the number of supposed new employees whose skills are under the

standard skill level. The other input is the percentage ratio of the skills of new employees compared to the standard value which is 100%. Table 4 shows the simulation results with the three pre-selected ratios (75%, 85% and 95%).

Efficiency. of new empl.	Avg. tard. (h)	Max tard. (h)	WIP
75%	5.73	27.20	45
85%	3.25	16.39	15
95%	0.98	5.15	1

Table 4: Simulated effect of differing worker efficiency levels on average and maximal tardiness value.

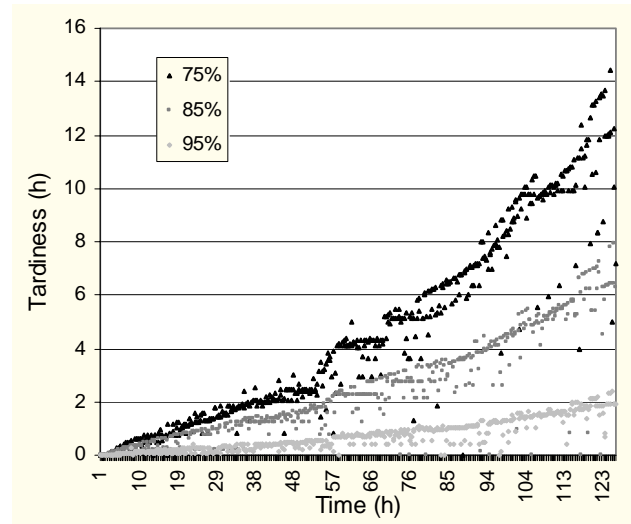


Figure 8: The effect of new employees.

Figure 8 presents the effect of differing worker efficiency levels on schedule execution. Running the simulation for a longer time horizon, the effect of the learning curve regarding new employees on the calculated schedule can be analysed easily.

### 4.2 Simulation-based scheduling

The simulation model presented above is applied as an evaluator of the short-time scheduler. As described previously, in the above model only one specific dispatching rule is applied, guaranteeing the sequential play-back of the schedule calculated by the constraint-based scheduler. Nevertheless, the same simulation model can be applied as schedule generator. Using the simulation in this way, predefined control rules should be added to the simulation model that provide the control logic of the simulation runs. Contrary to the previous application, the main input of this application mode is the list of production orders while the output is represented by the schedules generated by different simulation runs (Figure 9).

The other inputs of the system, such as process plans, routings, shift descriptions etc. reside further in the production database providing this way the base for the same automatic model-building procedure as presented in the previous section.

Naturally, the uncertainty factors, too, are considered as variable inputs of the simulation-based scheduler, like in the previous model. One of the most important requirements for this application is the fast response of the simulation model, to ensure enough simulation replications.

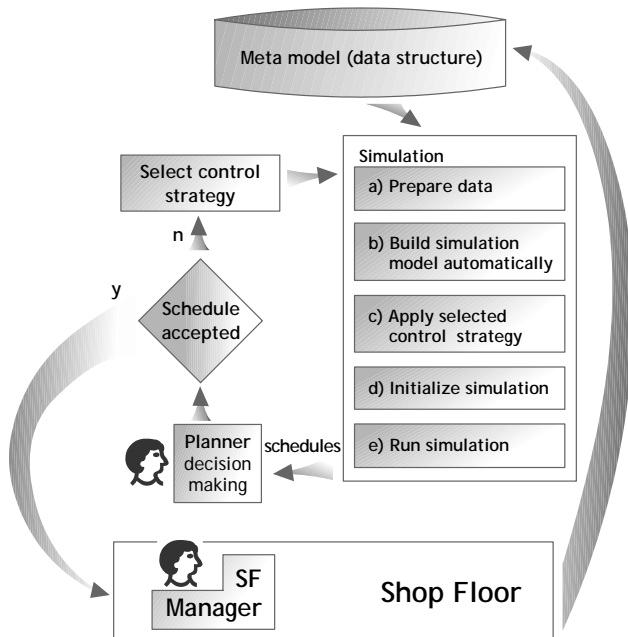


Figure 9: The methodology of applying the simulation model as a schedule generator.

## 5 CONCLUSION

The paper presented a hierarchical PPS system whose components - a medium-term aggregate capacity and production planner, a short-term job-shop scheduler, and a discrete-event simulator - work on the more and more detailed models of a given production environment. While the basic models of the planner and scheduler are deterministic, the simulator can capture non-deterministic events - especially the ones that may occur on the shop-floor.

The paper gave a detailed description on the architecture and the functionalities of the simulation module. Different experiments can be performed with the overall system supporting the mid- and short-term planning and control decisions on the shop-floor.

Some possible roles of simulation were also discussed in the first part of the paper, together with a brief introduction on the eventual evaluation of short-term schedules.

Parallel to the quasi-emulation mode, the presented simulation model can be applied traditionally as a simulation-based scheduler. In this case predefined control algorithms should be included in the simulation model and these predefined control decisions can be evaluated with the model.

## 6 ACKNOWLEDGEMENT

The research was supported partially by the project "Digital enterprises, production networks" in the frame of the National Research and Development Programme by the Ministry of Education (Grant No. 2/040/2001). A part of the work was covered by the National Research Fund, Hungary, Grant Nos. T034632 and T043547. Botond Kádár greatly acknowledges the support of Bolyai János Scholarship of the Hungarian Academy of Sciences, significantly easing the contribution.

## 7 REFERENCES

[1] E. Westkämper, D. Jovanoski, T. Rist: *New Framework for Digital Factory Planning*, [http://www.e-industrial-services.de/servlet/is\\_/12641/new-framework-for-digital-factory-planning.pdf](http://www.e-industrial-services.de/servlet/is_/12641/new-framework-for-digital-factory-planning.pdf) (visited: 23 March 2004)

[2] Pfeiffer, A., Kádár, B., Monostori, L., 2003, Evaluating and improving production control systems by using emulation, Proc. of the 12 IASTED Int. Conf. on Applied Simulation and Modelling, ASM 2003, 3-5 September, Marbella, Spain:261-267.

[3] Mc Gregor, I., The relationship between simulation and emulation, 2002, Proceedings of the 2002 Winter Simulation Conference, San Diego, California, USA:1683-1688

[4] Kovács, A., Kis, T., 2003, Partitioning of trees with respect to new criteria, Information Processing Letters, Submitted paper.

[5] Smith, S.F., 1994, Intelligent Scheduling, Chap, OPIS: A Methodology and Architecture for Reactive Scheduling, Morgan Kaufmann, San Francisco.

[6] Szelke, E., Monostori, L., 1999, Modeling Manufacturing Systems, Chap., Reactive scheduling in real-time production control, Springer, Berlin, Heidelberg, New York:65-113.

[7] McKay, K.N., Wiers, V.C.S., 1999, Unifying the theory and practice of production scheduling, Journal of Manufacturing Systems, 18/4:241-255.

[8] Bongaerts, L., 1998, Integration of scheduling and control in holonic manufacturing systems, PhD thesis, Catholic University Leuven, Belgium.

[9] Kádár, B., 2002, Intelligent approaches to manage changes and disturbances in manufacturing systems, PhD thesis, Technical University of Budapest, Hungary.

[10] Sadeh, N., Fox, M.S., 1990, Variable and value ordering heuristics for activity based job-shop, Proceedings of the Fourth Int. Conference on Expert Systems in Production and Operation Management, Hilton Head Island, USA:134-144.

[11] Berry, P.M., 1992, Scheduling: A problem of decision making under uncertainty, Proc. of 10th European Conf. on AI, ECAI'92, Vienna, Austria:638-642.

[12] Monostori, L., Márkus, A., Van Brussel, H., Westkamper, E., 1996, Machine learning approaches to manufacturing, CIRP Annals 45/2:675-712.

[13] Kempf, K., Uzsoy, R.; Smith, S.; Gary, K., 2000, Evaluation and comparison of production schedules, Computers in Industry, 42:203-220.

[14] Kis, T., 2002, A branch-and-cut algorithm for scheduling projects with variable-intensity activities, Mathematical Programming, Submitted paper.

[15] Kovács, A., Váncza, J., Kádár, B., Monostori L., Pfeiffer, A., 2003, Real life scheduling using constraint programming and simulation, Proc. of IMS 2003, 7th IFAC Wshop on Intelligent Manufacturing Systems, Budapest, Hungary:233-239.

[16] Monostori, L.; Váncza, J.; Márkus, A.; Kádár, B.; Viharos Zs.J., 2003, Towards the Realization of Digital Enterprises, 36th CIRP International Seminar on Manufacturing Systems, June 03 - 05, Saarland University, Saarbrücken, Germany:99-106